A short introduction to Stata

Course 1152M: Intellectual Capital and Knowledge Systems
Trudie Schils, April 16 2009, Maastricht University

The basics

Stata is a software package for statistical analyses, as typically used in academic research. It is a powerful and yet easy to use statistical package that runs on Windows, Macintosh and Unix platforms. Alternative programs used by economists are SPSS, TSP, or SAS. Because of the size of the data sets that will be analyzed in this course, and the nature of the analyses, it is impossible to use spreadsheet software such as Microsoft Excel. Stata is an excellent tool for data manipulation: moving data from external sources into the program (*e.g.* from Microsoft excel), generating new variables, generating summary data, merging data sets etc. Consequently, Stata can be used for answering ad hoc questions about any aspect of the data and this will be the main task within this course. To carry out the empirical assignments required for this course, ready-to-use Stata data files are made available in EleUM, containing a set of variables that can be used for doing the empirical part of the respective assignments.

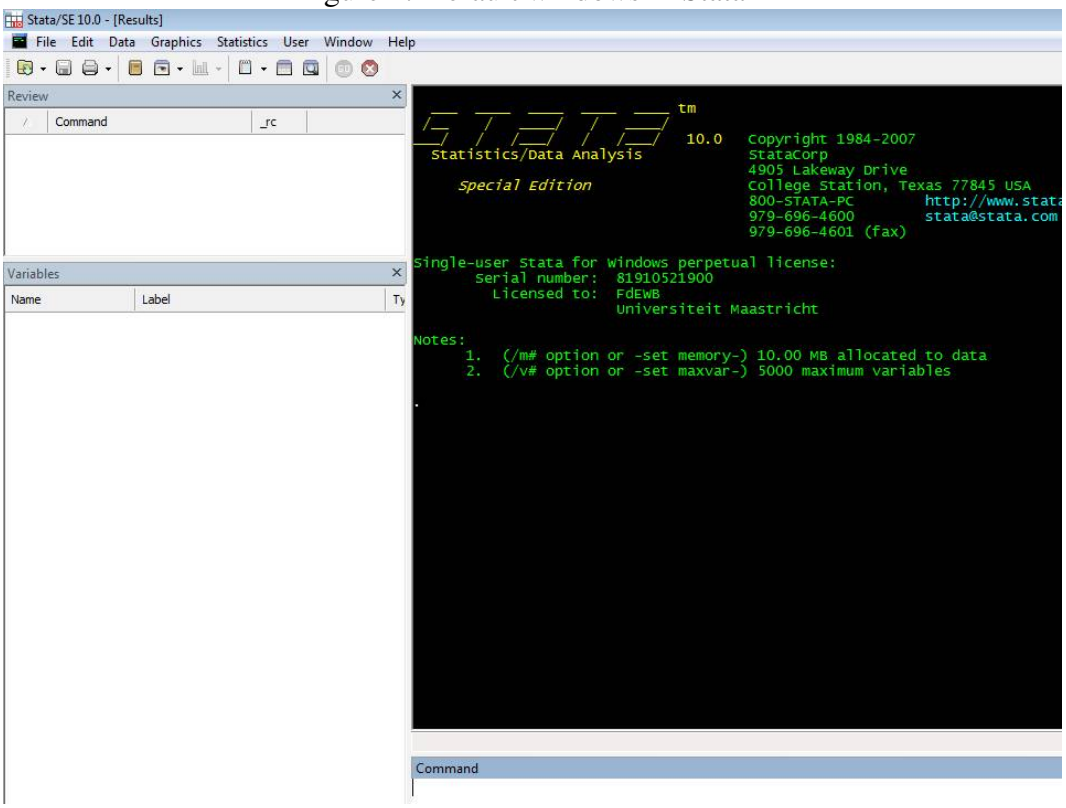
Three types of files are important to remember when working with Stata. Data files in Stata format can be recognized by the ".dta" extension. Of further importance are the do-files, that are recognized by the ".do" extension. Finally, ".smcl" files are log files in which your activities are stored, when you tell Stata to do so.

The best way to work in Stata is to make a do-file. A do-file contains all your manipulations with the data and you can keep it on your computer. Consequently, by using a do-file you (i) can always reconstruct what you have done; (ii) do not need to repeat the same manipulations again and again; and (iii) can easily adjust analyses by making slight changes to the do-file. A do-file differs from the log file (*i.e.* the .smcl file) because in the latter all responses to your commands are given, including possible mistakes you made. The log file is just for keeping track of your actions and the results, whereas a do-file is the main file that drives your

analyses.

To begin, start Stata, which is included among the programs. Stata will start, showing four windows, as shown in figure 1. On the top left you find the 'Review" window that shows the latest commands you used. Below this window you find the "Variables" window in which you can see the list with variables included in the data file. At the right you find the largest window, "Results", in which the results are shown. Finally, in the bottom you find the "Command" window in which you type your commands.

Figure 1: Default windows in Stata



Open a do-file using the toolbar "New Do-file Editor". You can start the do-file with some text that allows you to remember what this do-file is about and when you have created it, so you remember this in the future. For example, type

```
/* File created by Trudie Schils, April 16 2009 */
/* Background analyses for Task 1 of Course 1152M */
```

Using the `/*` and `*/` means that Stata recognizes everything in between as text, rather than a command.

The first thing of importance is to tell Stata in what directory you will be working, where your data and do-files *etc.* are to be stored. The command for this is

```
cd c:\data\1152M
```

(or any other directory of your choice)

Then save the do-file in your working directory, but keep it open. You can run the do-file by selecting the commands in your do-file and clicking "Do selected lines" in the toolbar (last icon at the right). The commands are executed, which you can observe in the "Results" window.

The next step is to open a log file to keep track of all your actions and results of these actions. This is done as follows:

```
log using Task1_1
```

which saves the file Task1_1.smcl in your working directory. The next time you use Stata to work on Task 1 you are recommended to call the new log file something like Task1_2, so you always end up with a nice order of Stata log files that show what you have been doing.

An example

As an example, we analyze data of the International Adult Literacy Survey (IALS). First save the file "ials.dta" on your computer in the directory of your choice (as specified above). The IALS data set contains information about a representative sample of adults (age 15 and older) in several countries. Open the "ials.dta" file, by typing

```
use ials.dta
```

You will see the variables included in the file in the "Variables" window. The file contains country, sex, age, hours, wage, years of schooling (yos), level of education (educat), literacy score, and whether the person is born in the country of residence (born). When you click on the "Data browser" in your toolbar, a new screen appears showing the data values. You can see that the first person in the sample is from Hungary (country=25), male, aged 33, working 35 hours a week, with 20,000 annual earnings, who completed 11 years of schooling, equivalent to ISCED level 3 (ISCED is an internationally accepted index for education levels), with a score of 312.9992 on the literacy test (all persons in the sample took a literacy test) and he is born in the country of residence, in this case Hungary. Close the "Data Browser".

If you want to get some descriptive statistics, *e.g.* you want to know how many men and women there are in the data, type
`ta sex` (`ta` is short for `tabular` showing the frequency of a variable)
and Stata produces a table, from which you can learn that there are 10,718 men and 9,670 women in the sample. The relative and cumulative frequencies are also shown in the table.

If you want to know more about specific commands in Stata, type `help tabular` and Stata will produce text that clarifies the command and the use of it. In general, the Help-function of Stata can be of use. By scrolling down the help menu you can learn the basics of Stata within a short time span.

For finding more detailed descriptive statistics on certain variables, you use the `tabstat` command. An important option with this command is the `tabstat, stat(<detail>)` option, with which you specify the descriptives you need. Details include mean (mean), standard deviation (sd), number of cases (n) median (med or p50), variance (var), minimum (min), maximum (max), and specified percentiles (p10 - p90). (Use help - search - tabstat for more options if necessary).

For example, if you want to know the mean wage and the standard deviation by country, type:
`tabstat wage, stat(mean,sd) by(country)`
and you find that the mean wage in the German sample is 30,987.98 (s.d.=19,124),

in the Netherlands 45,165.52 (s.d.=27,534), and in the United States 30,669.57 (s.d.=23,835). (note that these wages are in the country's own currencies).

To calculate the 10th-percentiles and 90th-percentile of the wage distribution you use:

```
tabstat wage, stat(p10, p90) by(country)
```

and you find that the 10th-percentile of the wage distribution of the complete dataset equals 18,000 and the 90th-percentile 550,000. More interesting are the country differences. In Germany the 10th-percentile equals 9,000 and the 90th-percentile equals 15,000, while these equal 7,500 and 77,500 respectively in the Netherlands.

Cross-tabulations can also be very useful. An example is this table providing the fractions of men and women in each country:

*Table 1: COUNTRY * SEX Crosstabulation*

	Sex		
Country	male	female	Total
Germany	54.09	45.91	100.00
Usa	52.51	47.49	100.00
Netherlands	55.24	44.76	100.00
Poland	50.13	49.87	100.00
Sweden	50.42	49.58	100.00
Italy	55.23	44.77	100.00
Norway	52.91	47.09	100.00
Slovenia	50.00	50.00	100.00
Czech republic	45.57	54.43	100.00
Denmark	54.99	45.01	100.00
Finland	50.68	49.32	100.00
Hungary	47.29	52.71	100.00
Canada	52.34	47.66	100.00
Switzerland	52.29	47.71	100.00
Chile	63.19	36.81	100.00
Total	52.57	47.43	100.00

which we retrieved using the command:

```
ta country sex, ro nof
```

With “ro” (short for row) we tell Stata to calculate row percentages (use column or co to get column percentages) and “nof” is used to suppress the absolute values.

Data manipulation

Sometimes data have to be manipulated before you can perform your statistical analyses. For example, economists usually investigate the log of the wage rather than the wage itself to be able to tell something about the percentage change in wages rather than the wage changes in dollars (or euros). You can create a new variable “lnw” using the command:

```
gen lnw = ln(wage)
```

In the “Variables” window you will now see the new variable at the bottom of the list. If you want to label the variable, you can use:

```
lab var lnw "Log wages"
```

and you will see this name showing up in the “Variables” window.

Hourly wages can be generate by:

```
gen hw = wage / hours
```

And the log of the hourly wage then equals:

```
lhw = ln(hw)
```

Furthermore, it is possible to restrict the analyses to a subgroup of the sample:

```
keep if country==11
```

throws away all countries except Sweden, while adding

```
if country==11
```

to your commands keeps the other countries, but performs the commmand only for Sweden. For example

```
count if country==11
```

shows that the Swedish sample contains 1545 observations.

Simple regression analysis

Selecting only data for men aged over 35 years in Sweden, it can be investigated how much a year of education on average contributes to earnings of men in Sweden. To analyze this, you can do a simple regression:

```
reg lnw yos if country==11 & sex==0 & age>=35
```

Such a regression is called an ordinary least squares (OLS) regression and yields the following output:

Source	SS	df	MS		Number of obs	=	542
					F(1, 540)	=	40.96
Model	11.1132988	1	11.1132988		Prob > F	=	0.0000
Residual	146.508458	540	.271311959		R-squared	=	0.0705
					Adj R-squared	=	0.0688
Total	157.621757	541	.291352699		Root MSE	=	.52088

lnw	Coef.	Std. Err.	t	$P>\mid t \mid$	[95% Conf.	Interval]
yos	.036259	.0056654	6.40	0.000	.0126927	.0473877
cons	11.75017	.0694697	169.14	0.000	11.51632	11.88663

This means that in Sweden for men aged over 35 years:
$ln(wage) = 11.750 + .036$**years of schooling + error term*
So, in Sweden every year in school is associated with approximately 3.6% higher earnings.

The standard error informs you about the uncertainty of these estimates. At at confidence-level of 5% the true parameter of years of schooling is in the interval between $.036 - 1.96 * .006$ and $.036 + 1.96 * .006$. When 0 is not in this interval, we say that the parameter is significant, or that the parameter is significantly different from zero.

t = Coef./Std.Error. A t-value below -1.96 or above 1.96 implies that the parameter is significant. The t-value of 6.248 we found for the returns to education in Sweden is rather high in this respect (which is likely due to the lack of other explanatory variables). The $P>\mid t \mid$ is another way to interpret significance, a more technical one. It shows the probability that the null hypothesis that the coefficient

is zero (no effect) is rejected. This null hypothesis is rejected when the p-value is small, given the level of significance one wishes to use, 1%, 5% or 10%. In this case, the p-value is 0.000, implying rejection of the null hypothesis at all levels of significance: the coefficient is significantly different from zero, implying a significant effect of years of schooling on log wages in Sweden.

Usually, when analyzing the relation between years of schooling and log wages, also sex, age and age squared are included as variables. Age squared is not available, so you have to create this variable using:
`gen agesq = age * age`
Then add the other explanatory variables to the regression command:
`reg lnw yos sex age agesq if country==11`
and the output will be as follows:

Source	SS	df	MS		Number of obs	=	1542
Model	145.287818	4	36.3219546		F(4,1537)	=	113.93
Residual	490.00484	1537	.318806012		Prob > F	=	0.0000
					R-squared	=	0.2287
					Adj R-squared	=	0.2267
Total	635.292658	1 541	.412259999		Root MSE	=	.56463

lnw	Coef.	Std. Err.	t	$P>\|t\|$	[95% Conf. Interval]	
yos	.0245991	.0040908	6.01	0.000	.016575	.0326232
sex	-.3253639	.0287937	-11.30	0.000	-.381843	-.2688847
age	.1103624	.0081085	13.61	0.000	.0944575	.1262672
agesq	-.0011431	.0000967	-11.83	0.000	-.0013326	-.0009535
cons	9.293897	.1667047	55.75	0.000	8.966904	9.620889

From this output you can learn that women in Sweden earn significantly less than men (32.5%), while earnings increase with age until the age of 55.

When you are ready, save the do-file, but **do not** save the data file. The next time you want to continue with your work, you can easily run the dofile again. When you make a mistake in the syntax, e.g. `gen wage = wage - hours`, your original data gets lost when saving the transformed data.

Dummies are often used in empirical analyses. For example, the variable sex in

the data set is a dummy variable. It can have the value 0 (male) or 1 (female). Whenever you want to regress sex (independent variable, dummy) on log wages (dependent variable), use the following commands:

```
reg lnw literacy sex
```

And you can see that the log wages are lower for women than for men.

Suppose you want to estimate the link between literacy and (log) wages using the data from all countries simultaneously, then

```
reg lnw literacy
```

would do the job. You will notice that every extra point on the literacy scale is associated with a .2% decrease in income. The problem is that the levels of the wages in the different countries can be very different, because different currencies are used and because the overall economic performance of these countries is very different. To take this into account for each country (except one so-called reference country) a dummy variable can be included. This dummy equals 1 for cases from this country and 0 otherwise. Stata has an easy way to use dummies in regression analysis, with the "xi" command. The regression, controlling for country differences, now becomes:

```
xi:  reg lnw literacy
```

You will find that Stata dropped one country dummy, because one country is taken as a reference. In this case country=5 is omitted (Germany), which you can see just above the regression results. The country dummies measure the difference from this reference country. As for the effect of literacy on log wages, you will now find that a one-point increase in the literacy scale is associated with a .22% increase in log wages.

Graphs

Stata also offers a wide range of graphs. Using the `histogram age` command you can plot the frequency distribution by age. Or you might produce a similar graph for literacy using `histogram literacy`.

You can also compare the literacy scale between two countries, using:
`histogram literacy if country==5 | country==6, by(country)`
The | represents 'or', and you are comparing Germany and the United States here. Stata will show you both graphs at the same time, allowing you to make inference from these graphs. Of course, comparing more countries at the same time is also possible.

Closing Stata

Whenever you are finished with this data you can clear the screen by typing:
`clear`
and you will notice that all windows are cleared. You then can close Stata by typing:
`exit`.

The data

The data set has been prepared, based on the raw data of the OECD International Adult Literacy Survey. The data set for this course, ials.dta, only includes data of the countries for which wage information is available. Only people with a job, *i.e.* with positive earnings, who are not self-employed, have been included. For some countries wage data are available in brackets only. Wage in the file is equal to the middle of the corresponding interval, or 1.4 times the lower bracket for the top-incomes interval.

For Sweden, the number of hours worked per week is not available. The hours of part-timers therefore have been set equal to 18. For Canada age is only available in brackets, therefore the middle is taken again, *e.g.* for all workers in the interval 16 to 25, the age is set at 20.